Natural Main Menu

The Natural Main Menu provides access to Natural development functions, environment settings, utilities and example libraries.

The section below contains information on the functions and input options provided with the Natural Main Menu and its subordinate menus:

- Invoking the Natural Main Menu
- Development Functions
- Development Environment Settings
- Maintenance and Transfer Utilities
- Debugging and Monitoring Utilities
- Example Libraries
- Other Products

Invoking the Natural Main Menu

There are two methods of invoking the Natural Main Menu:

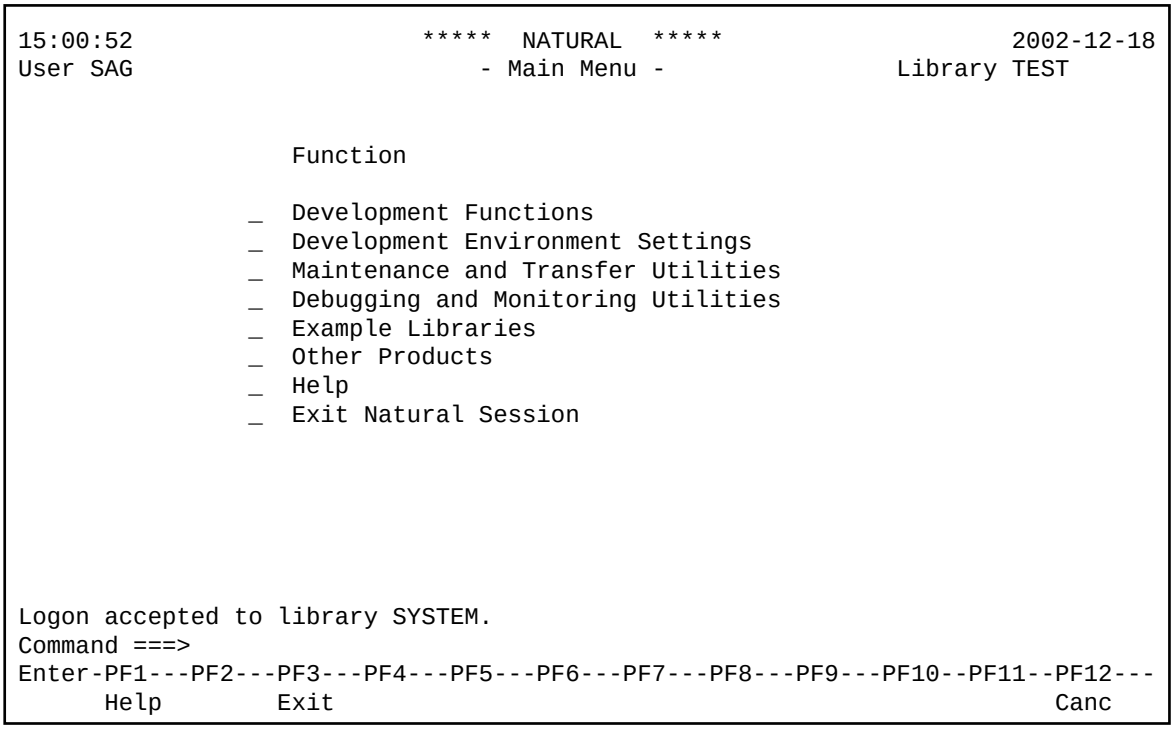
- You can define a default setting by switching on Menu mode. Menu mode causes the Natural Main Menu to be invoked automatically for the next session started.
- You can invoke the Natural Main Menu within a Natural session whenever desired.

To switch Menu mode on or off

- At Natural startup, specify the Natural parameter MENU=ON (activate), or MENU=OFF (deactivate).
  See also MENU in the Natural Parameter Reference documentation.

To invoke the Natural Main Menu within a Natural session

- At a Natural command prompt (such as NEXT or MORE), enter the system command MAINMENU.
- The Natural Main Menu appears:



See also Executing a Menu Function in the section Executing Commands and Menu Functions.

Each function listed on the Natural Main Menu invokes a menu of the corresponding name where you can select further functions:

Function on Natural Main Menu	**Explanation of Functions on Corresponding Menu**
Development Functions	Create and maintain programs, maps, data areas and other components that make up a Natural application.
Development Environment Settings	Display and modify various settings that affect your Natural session.
Maintenance and Transfer Utilities	Invoke a Natural utility to create and maintain certain objects or transfer them to another environment.
Debugging and Monitoring Utilities	Invoke a Natural utility to monitor your Natural applications and locate errors in their processing flow.
Example Libraries	Select libraries containing example programs and application programming interfaces (APIs).
Other Products	Invoke other Software AG products.

Note:
The position and color of the message line and PF-key lines on the Natural Main Menu and its subordinate menus can be changed with the user exit USR2003P (provided in the Natural system library SYSEXT).

Development Functions

The functions listed on the Development Functions menu are those you will need most frequently when you develop an application with Natural. The functions apply to all Natural programming objects and Predict Descriptions (if Predict is installed) that are available in the library where you are currently logged on.

Other Related Topics:

- Natural Programming Guide
- Executing Commands and Menu Functions

The table below contains information on the fields provided on the Development Functions menu:

Field	Explanation
User	The ID of the Natural user who logged in the current session.
Library	The Natural library currently active. See also how to switch libraries in Switching Natural Libraries.
Mode	The programming mode: reporting or structured mode. See Programming Modes below.
Work area empty	Indicates that no source has been loaded into the Natural work area. If an source has already been loaded into the work area, the type and the name of the object will be displayed instead, for example, Program PROGX.
Code	The code that corresponds to the function desired, for example, **C** for Create Object. See also Executing a Menu Function in the section Executing Commands and Menu Functions.
Type	The type of programming object, such as **P** for Program. For further details, see Object Types in the Natural Programming Guide. You can leave the Type field blank if you specify the name of a Natural object that already exists.
Name	The name of the programming object. For an explanation of valid object names, see Object Names (General Information) in the Natural Editors documentation.
Command line	The Command line is an input field in which you enter a Natural command. For example: To edit an existing program named PROGX, in the Command line, you would enter the system command: `EDIT PROGX` See also Executing Commands and Menu Functions.
PF Keys	The PF keys can be used as an alternative to using system commands or menu functions. The PF-key lines at the bottom of the screen indicate which function is assigned to which key. See also Standard PF Keys in Executing Commands and Menu Functions

The table below contains information on the functions provided on the Development Functions menu. For most of the menu functions, there are equivalent Natural system commands. These alternative system commands are listed in the table and further explained in the relevant sections in the Natural System Commands documentation.

Function	Code	Explanation
Create Object	C	Invokes a Natural editor where you can create a new programming object, such as a program, map or data area. Specify the type and the name of the object to be created. You can enter a question mark (?) in the Type field, to select an object type from a list of all types available for this function.
Edit Object	E	Invokes a Natural editor and displays the source of the specified programming object in modify mode. Specify the name of an existing object to be edited. You can also invoke a selection list of objects: see Specifying Object Ranges below. You can enter a question mark (?) in the Type field, to select an object type from a list of all types available for this function. Equivalent system command: EDIT
Rename Object	R	Invokes the Rename Objects window where you change the name of the specified programming object and/or the object type. Equivalent system command: RENAME
Delete Object	D	Invokes the Delete window for the specified programming object. In the Delete window, confirm the deletion by entering the name of the object again in the relevant input field. You can also invoke a selection list of objects as described in Specifying Object Ranges below. On this list, you can mark one or more objects for deletion. Equivalent system command: DELETE
Execute Program	X	Executes a programming object of the type Program. Specify the name of the programming object to be executed. Other object types cannot be executed by themselves, but must be invoked from another object. Equivalent system command: EXECUTE
List Object(s)	L	Displays the source of the specified programming object. Specify the name of the programming object to be displayed. You can also invoke a selection list of objects: see Specifying Object Ranges below. You can enter a question mark (?) in the Type field, to select an object type from a list of all types available for this function. Equivalent system command: LIST
List Subroutines Used	S	Ascertains which programming objects use which external subroutines and classes. Equivalent system command: ROUTINES

The section below contains information on:

- Programming Modes
- Natural Editors
- Specifying Object Ranges

Programming Modes

Natural offers two programming modes: reporting mode and structured mode. Generally, it is recommended to use structured mode exclusively, because it provides for more clearly structured applications. Therefore, all explanations and examples in the documentation Natural for Mainframes - Tutorial and Natural Editors refer to structured mode. Any peculiarities of reporting mode will not be taken into consideration.

In the top right-hand corner of the Development Functions menu is the Mode field, which indicates the programming mode currently in effect: Structured or Reporting.

To switch programming modes

- In the Mode field, overwrite the first position with an **S** to switch on structured mode, or an **R** to switch on reporting mode.

  Or, in the Command line, enter the system command GLOBALS SM=ON to switch on structured mode, or GLOBALS SM=OFF to switch on reporting mode. See also GLOBALS in the Natural System Command Reference documentation.

Related Topic:

- Reporting Mode and Structured Mode in the Natural Programming Guide.

Natural Editors

Depending on the type of the programming object, Natural invokes the appropriate editor: the program editor, the map editor or the data area editor. For details, see the relevant sections in the Natural Editors documentation.

Specifying Object Ranges

With the functions Edit Object, List Object(s) and Delete Object, you can specify the name of a single programming object or a range of objects. Specifying a range of objects will generate a list from which you can select one or more objects you wish to edit or list, or mark for deletion.

You can list either all programming objects available in the current library, or objects with names that start with a certain value.

To list all objects

- In the Name field, enter an asterisk (*).
  You will receive a list with all programming objects available in the current library.

To list objects with start values

- In the Name field, specify a start value followed by an asterisk (*).
  This option to enter a value followed by an asterisk is referred to as asterisk notation.
  Example:

  ```
  AB*
  ```

  Selects all objects that begin with AB, such as AB, AB1, ABC, ABEZ.
  Does not select objects that start with AA1 or ACB, for example.

Note:
The List Object(s) function provides further options to specify object ranges as described for the equivalent system command LIST.

Development Environment Settings

The table below contains a brief description of the functions provided on the Development Environment Settings menu, and lists the Natural system commands that correspond to these functions. For details on a system command, refer to the relevant section in the Natural System Command Reference documentation.

Function	Explanation	Correspond. Command
Function-Key Settings	Assigns functions to PF keys to be used in your Natural session.	KEY
Compilation Settings	Sets options that affect the way in which Natural programming objects are compiled.	COMPOPT
Session Parameter Settings	Changes the settings of Natural session parameters. Session parameters are described in the Natural Parameter Reference documentation.	GLOBALS
Profile Parameter Settings	Changes the settings of Natural profile parameters. Profile parameters are described in the Natural Parameter Reference documentation and in Profile Parameter Usage in the Natural Operations for Mainframes documentation. The system command SYSPARM invokes a utility of the same name that is described in the Natural Utilities documentation.	SYSPARM
Technical Session Information	Displays technical information on your Natural session, such as user ID and operating system.	TECH
System File Information	Displays the current definitions of the Natural system files. See also: Natural System Files in Natural Components.	SYSPROF
Product Installation Information	Displays a list of the products installed at your site and information on these products.	SYSPROD
Security Profile Information	Only available if Natural Security is installed. Displays the security profile currently in effect for you.	PROFILE

Maintenance and Transfer Utilities

The table below contains a brief description of the functions provided on the Maintenance and Transfer Utilities menu, and lists the Natural system commands that correspond to these functions. Each of these commands invokes a Natural utility that is described in the Natural Utility documentation.

Function	Explanation of Utility	Correspond. Command
Maintain Error Messages	Creates and maintains messages you wish to issue in your Natural applications.	SYSERR
Maintain DDMs	Creates and maintains data definition modules (DDMs), that is, logical definitions of the database files you wish to access in your Natural applications. For a detailed explanation of DDMs, see the section Database Access in the Natural Programming Guide.	SYSDDM
Maintain Command Processors	Creates and maintains the command processors you wish to use in your Natural applications.	SYSNCP
Maintain Remote Procedure Calls	Creates and maintains remote procedure calls, that is, provides the settings necessary to execute a Natural subprogram located on a remote server.	SYSRPC
Transfer Objects to Other Libraries	Transfers Natural programming objects, error messages, DDMs and several other objects from one library to another.	SYSMAIN
Transfer Objects to Other System Files	Unloads or loads Natural programming objects. You can use either the system command SYSUNLD to invoke the initial utility menu for unloading or loading objects, or the system command NATUNLD or NATLOAD to directly invoke the subordinate load or unload utility: NATUNLD utility: unloads Natural programming objects, error messages and DDMs from a Natural system file to a work file. NATLOAD utility: loads Natural programming objects, error messages and DDMs from a work file into a Natural system file.	SYSUNLD
Transfer Objects to Other Platforms	Transfers Natural programming objects, DDMs, error messages and Adabas FDTs from one hardware platform to another.	SYSTRANS
Transfer Objects to Other Systems	Invokes the Object Handler to process Natural and non-Natural objects for distribution in Natural environments.	SYSOBJH

Debugging and Monitoring Utilities

The table below contains a brief description of the functions provided on the Debugging and Monitoring Utilities menu, and lists the Natural system commands that correspond to these functions. Each of these commands invokes a Natural utility that is described in the Natural Utility documentation.

Function	Explanation of Utility	Correspond. Command
Debugging	Searches for errors in the processing flow of programs.	TEST
Logging of Database Calls	Logs database commands.	TEST DBLOG
Issuing Adabas Calls	Passes Adabas commands directly to the database.	SYSADA
Buffer Pool Maintenance	Monitors the Natural buffer pool and adjusts it to meet your requirements.	SYSBPM
Editor Buffer Pool Maintenance	Monitors the buffer pool of the Software AG Editor and adjusts it to meet your requirements.	SYSEDT
TP-Specific Monitoring	Monitors and controls TP-monitor-specific characteristics of Natural.	SYSTP
Data Collection and Tracing	Collects monitoring and accounting data about the processing flow of a Natural application.	SYSRDC
Error Information on Abnormal Termination	Provides information Software AG technical support requires for error diagnosis.	DUMP

Example Libraries

When you select Example Libraries from the Natural Main Menu, a list of libraries is displayed. These libraries contain example programs for demonstration purposes and application programming interfaces (APIs) provided by Software AG:

Library	Contents
SYSEXPG	Example programs shown and referred to in the Natural Programming Guide.
SYSEXRM	Example programs shown and referred to in the Natural Statements documentation and the Natural System Variables documentation.
SYSEXV	Example programs that illustrate new Natural features.
SYSEXT	Application programming interfaces (APIs). See also the system command SYSEXT as described in the Natural System Command Reference documentation.
SYSEXTP	Example programs and APIs for specific functions that apply only under certain TP monitors.

Other Products

When you select Other Products from the Natural Main Menu, a list of Software AG add-on products appears. These products are installed at your site and can be accessed from this menu.

